A Characterization of Combined Traces Using Labeled Stratified Order Structures

Dai Tri Man Lê

Department of Computer Science, University of Toronto
10 King's College Road, Toronto, ON, M5S 3G4 Canada
ledt@cs.toronto.edu

Abstract. This paper defines a class of labeled stratified order structures that characterizes exactly the notion of *com*bined *traces* (i.e., *comtraces*) proposed by Janicki and Koutny in 1995. Our main technical contributions are the representation theorems showing that comtrace quotient monoid, *combined dependency graph* (Kleijn and Koutny 2008) and our labeled stratified order structure characterization are three different and yet equivalent ways to represent comtraces.
Keywords. causality theory of concurrency, combined traces monoids, step sequences, stratified order structures, label-preserving isomorphism.

1 Introduction

Partial orders are a principle tool for modelling "true concurrency" semantics of concurrent systems (cf. [29]). They are utilized to develop powerful partial-order based automatic verification techniques, e.g., *partial order reduction* for model checking concurrent software (see, e.g., [1, Chapter 10] and [8]). Partial orders are also equipped with *traces*, their powerful formal language counterpart, proposed by Mazurkiewicz [28]. In *The Book of Traces* [5], trace theory has been used to tackle problems from diverse areas including formal language theory, combinatorics, graph theory, algebra, logic, and *concurrency theory*.

However, while partial orders and traces can sufficiently model the "earlier than" relationship, Janicki and Koutny argued that it is problematic to use a single partial order to specify both the "earlier than" and the "not later than" relationships [17]. This motivates them to develop the theory of *relational structures*, where a pair of relations is used to capture concurrent behaviors. The most well-known among the classes of relational structures proposed by Janicki and Koutny is the class of *stratified order structures* (*so-structures*) [12,16,18,19,15]. A so-structure is a triple $(X, \prec, \sqsubset)$, where $\prec$ and $\sqsubset$ are binary relations on X. They were invented to model both the "earlier than" (the relation $\prec$) and "not later than" (the relation $\sqsubset$) relationships, under the assumption that system runs are described by *stratified partial orders*, i.e., step sequences. They have been successfully applied to model inhibitor and priority systems, asynchronous races, synthesis problems, etc. (see for example [18,20,24,23,25,26] and others).

The *com*bined *trace* (*comtrace*) notion, introduced by Janicki and Koutny [18], generalizes the trace notion by utilizing step sequences instead of words. First the set of all possible steps that generates step sequences are identified by a relation *sim*, which is

called *simultaneity*. Second a congruence relation is determined by a relation *ser*, which is called *serializability* and in general *not* symmetric. Then a comtrace is defined as a finite set of congruent step sequences. Comtraces were introduced as a formal language representation of so-structures to provide an operational semantics for Petri nets with inhibitor arcs. Unfortunately, comtraces have been less often known and applied than so-structures, even though in many cases they appear to be more natural. We believe one reason is that the comtrace notion was too succinctly discussed in [18] without a full treatment dedicated to comtrace theory. Motivated by this, Janicki and the author have devoted our recent effort on the study of comtraces [21,27,22], yet there are too many different aspects to explore and the truth is we can barely scratch the surface. In particular, the huge amount of results from trace theory (e.g., from [5,6]) desperately needs to be generalized to comtraces. These tasks are often non-trivial since we are required to develop intuition and novel techniques to deal with the complex interactions of the "earlier than" and "not later than" relations.

This paper gives a novel characterization of comtraces using labeled so-structures. Such definition is interesting for the following reasons.

First, it defines exactly the class of labeled so-structures that can be represented by comtraces. It is worth noting that this point is particularly important. Even though it was shown in [18] that every comtrace can be represented by a labeled so-structure, the converse could not be shown because a class of labeled so-structures that defines precisely the class of comtraces was not known. The closest to our characterization is the *combined dependency graph* (*cd-graph*) notion (analogous to *dependence graph* representation of traces) introduced recently by Kleijn and Koutny [26], but again a theorem showing that combined dependency graphs can be represented by comtraces was not given. Our approach is quite different and based on some new ideas discussed in Section 4 of this paper.

Second, even though the step sequence definition of comtraces is more suitable when dealing with formal language aspects of comtraces, the labeled so-structure representation is more suitable for a variety of powerful order-theoretic results and techniques available to us (cf. [11,3,15]).

Finally, the labeled so-structure definition of comtrace can be easily extended to *infinite comtraces*, which describe nonterminating concurrent processes. The labeled poset representation of infinite traces is already successfully applied in both theory and practice, e.g., [31,9,10,14]. Although such definition is equivalent to the one using quotient monoid over infinite words [13,4], we believe that infinite labeled posets are sometimes simpler. Indeed the celebrated work by Thiagarajan and Walukiewicz (cf. [31]) on linear temporal logic for traces utilizes the labeled poset characterization of infinite traces, where *configurations* of a trace are conveniently defined as *finite downward closed subsets* of the labeled poset representation. We will not analyze infinite comtraces or logics for comtraces in this paper, but these are fruitful directions to explore using the results from this paper.

The paper is organized as follows. In Section 2, we recall some preliminary definitions and notations. In Section 3, we give a concise exposition of the theory of so-structures and comtraces by Janicki and Koutny [18,19]. In Section 4, we give our definition of comtraces using labeled so-structure and some remarks on how we arrived

at such definition. In Section 5, we prove a representation theorem showing that our comtrace definition and the one by Janicki and Koutny are indeed equivalent; then using this theorem, we prove another representation theorem showing that our definition is also equivalent to the cd-graph definition from [26]. In Section 6, we define *composition* operators for our comtrace representation and for cd-graphs. Finally, in Section 7, some final remarks and future works are presented.

2   Notations

2.1   Relations, Orders and Equivalences

The *powerset* of a set X will be denoted by $\wp(X)$, i.e. $\wp(X) \triangleq \{Y \mid Y \subseteq X\}$. The set of all *non-empty* subsets of X will be denoted by $\wp^{\setminus\{\emptyset\}}(X)$. In other words, $\wp^{\setminus\{\emptyset\}}(X) \triangleq \wp(X) \setminus \{\emptyset\}$.

We let id_X denote the *identity relation* on a set X. If R and S are binary relations on a set X (i.e., $R, S \subseteq X \times X$), then their *composition* $R \circ S$ is defined as $R \circ S \triangleq \{(x,y) \in X \times X \mid \exists z \in X.\ (x,z) \in R \land (z,y) \in S\}$. We also define

$$R^0 \triangleq id_X \qquad R^i \triangleq R^{i-1} \circ R \quad (\text{for } i \geq 1) \qquad R^+ \triangleq \bigcup_{i \geq 1} R^i \qquad R^* \triangleq \bigcup_{i \geq 0} R^i$$

The relations R^+ and R^* are called the *(irreflexive) transitive closure* and *reflexive transitive closure* of R respectively.

A binary relation $R \subseteq X \times X$ is an *equivalence relation* relation on X if and only if (iff) R is *reflexive*, *symmetric* and *transitive*. If R is an equivalence relation, then for every $x \in X$, the set $[x]_R \triangleq \{y \mid y\, R\, x \land y \in X\}$ is the equivalence class of x with respect to R. We also define $X/R \triangleq \{[x]_R \mid x \in X\}$, i.e., the set of all equivalence classes of X under R. We drop the subscript and write $[x]$ when R is clear from the context.

A binary relation $\prec \subseteq X \times X$ is a *partial order* iff R is *irreflexive* and *transitive*. The pair $(X, \prec)$ in this case is called a *partially ordered set* (*poset*). The pair $(X, \prec)$ is called a *finite poset* if X is finite. For convenience, we define:

$$\simeq_\prec \triangleq \big\{(a,b) \in X \times X \mid a \not\prec b \,\land\, b \not\prec a\big\} \qquad\qquad (incomparable)$$
$$\frown_\prec \triangleq \big\{(a,b) \in X \times X \mid a \simeq_\prec b \,\land\, a \neq b\big\} \qquad (distinctly\ incomparable)$$
$$\prec^\frown \triangleq \big\{(a,b) \in X \times X \mid a \prec b \,\lor\, a \frown_\prec b\big\} \qquad\qquad (not\ greater)$$

A poset $(X, \prec)$ is *total* iff $\frown_\prec$ is empty; and *stratified* iff $\simeq_\prec$ is an equivalence relation. Evidently every total order is stratified.

2.2   Step Sequences

For every finite set X, a set $\mathbb{S} \subseteq \wp^{\setminus\{\emptyset\}}(X)$ can be seen as an alphabet. The elements of $\mathbb{S}$ are called *steps* and the elements of $\mathbb{S}^*$ are called *step sequences*. For example, if the set of possible steps is $\mathbb{S} = \big\{\{a,b,c\},\{a,b\},\{a\},\{c\}\big\}$, then $\{a,b\}\{c\}\{a,b,c\} \in \mathbb{S}^*$ is a step sequence. The triple $(\mathbb{S}^*, *, \varepsilon)$, where $\_ * \_$ denotes the step sequence concatenation operator (usually omitted) and ε denotes the empty step sequence, is a monoid.

Let $t = A_1 \ldots A_k$ be a step sequence. We define $|t|_a$, the number of occurrences of an event a in w, as $|t|_a \triangleq \big|\{A_i \mid 1 \le i \le k \wedge a \in A_i\}\big|$, where $|X|$ denotes the cardinality of the set X. Then we can construct its unique *enumerated step sequence* $\bar{t}$ as

$$\bar{t} \triangleq \overline{A_1} \ldots \overline{A_k}, \text{ where } \overline{A_i} \triangleq \Big\{ e^{(|A_1 \ldots A_{i-1}|_e + 1)} \,\big|\, e \in A_i \Big\}.$$

We will call such $\alpha = e^{(j)} \in \overline{A_i}$ an *event occurrence* of e. For instance, if we let $t = \{a,b\}\{b,c\}\{c,a\}\{a\}$, then $\bar{t} = \{a^{(1)}, b^{(1)}\}\{b^{(2)}, c^{(1)}\}\{a^{(2)}, c^{(2)}\}\{a^{(3)}\}$.

We let $\Sigma_t = \bigcup_{i=1}^k \overline{A_i}$ denote the set of all event occurrences in all steps of t. For example, when $t = \{a,b\}\{b,c\}\{c,a\}\{a\}$, $\Sigma_t = \{a^{(1)}, a^{(2)}, a^{(3)}, b^{(1)}, b^{(2)}, c^{(1)}, c^{(2)}\}$. We also define $\ell : \Sigma_t \to E$ to be the function that returns the *label* of α for each $\alpha \in \Sigma_t$. For example, if $\alpha = e^{(j)}$, then $\ell(\alpha) = \ell(e^{(j)}) = e$. Hence, from an enumerated step sequence $\bar{t} = \overline{A_1} \ldots \overline{A_k}$, we can uniquely reconstruct its step sequence $t = \ell(\overline{A_1}) \ldots \ell(\overline{A_k})$.

For each $\alpha \in \Sigma_u$, we let $pos_t(\alpha)$ denote the consecutive number of a step where α belongs, i.e., if $\alpha \in \overline{A_i}$ then $pos_t(\alpha) = i$. For our example, $pos_t(a^{(2)}) = 3$, $pos_t(b^{(2)}) = pos_t(c^{(1)}) = 2$, etc.

It is important to observe that step sequences and stratified orders are interchangeable concepts. Given a step sequence u, define the binary relation $\lhd_u$ on Σ_u as

$$\alpha \lhd_u \beta \overset{df}{\iff} pos_u(\alpha) < pos_u(\beta).$$

Intuitively, $\alpha \lhd_u \beta$ simply means α occurs before β on the step sequence u. Thus, $\alpha \lhd_u^\frown \beta$ iff $(\alpha \ne \beta \wedge pos_u(\alpha) \le pos_u(\beta))$; and $\alpha \simeq_u \beta$ iff $pos_u(\alpha) = pos_u(\beta)$. Obviously, the relation $\lhd_u$ is a stratified order and we will call it the stratified order *generated by the step sequence u*.

Conversely, let $\lhd$ be a stratified order on a set Σ. The set Σ can be represented as a sequence of equivalence classes $\Omega_\lhd = B_1 \ldots B_k$ ($k \ge 0$) such that

$$\lhd = \bigcup_{i < j} B_i \times B_j \quad \text{and} \quad \simeq_\lhd = \bigcup_i B_i \times B_i.$$

The sequence $\Omega_\lhd$ is called the step sequence *representing* $\lhd$. A detailed discussion on this connection between stratified orders and step sequences can be found in [22].

3 Stratified Order Structures and Combined Traces

In this section, we review the Janicki – Koutny theory of stratified order structures and comtraces from [18,19]. The reader is also referred to [26] for an excellent introductory survey on the subject with many motivating examples.

3.1 Stratified Order Structures

A *relational structure* is a triple $T = (X, R_1, R_2)$, where X is a set and R_1, R_2 are binary relations on X. A relational structure $T' = (X', R_1', R_2')$ is an *extension* of T, denoted as $T \subseteq T'$, iff $X = X'$, $R_1 \subseteq R_1'$ and $R_2 \subseteq R_2'$.

Definition 1 (stratified order structure [19]). *A* stratified order structure *(so-structure) is a relational structure* $S = (X, \prec, \sqsubset)$, *such that for all* $\alpha, \beta, \gamma \in X$, *the following hold:*

    S1: $\alpha \not\sqsubset \alpha$ S3: $\alpha \sqsubset \beta \sqsubset \gamma \wedge \alpha \ne \gamma \implies \alpha \sqsubset \gamma$

    S2: $\alpha \prec \beta \implies \alpha \sqsubset \beta$ S4: $\alpha \sqsubset \beta \prec \gamma \vee \alpha \prec \beta \sqsubset \gamma \implies \alpha \prec \gamma$

When X is finite, S is called a finite so-structure. ■

The axioms S1–S4 imply that $\prec$ is a partial order and $\alpha \prec \beta \Rightarrow \beta \not\sqsubset \alpha$. The axioms S1 and S3 imply $\sqsubset$ is a *strict preorder*. The relation $\prec$ is called *causality* and represents the "earlier than" relationship while the relation $\sqsubset$ is called *weak causality* and represents the "not later than" relationship. The axioms S1–S4 model the mutual relationship between "earlier than" and "not later than" relations, provided that *the system runs are stratified orders*. Historically, the name "stratified order structure" came from the fact that stratified orders can be seen as a special kind of so-structures.

Proposition 1 ([17]). *For every stratified poset $(X, \lhd)$, the triple $S_{\lhd} = (X, \lhd, \lhd^{\frown})$ is a so-structure.* □

We next recall the notion of *stratified order extension*. This concept is extremely important since the relationship between stratified orders and so-structures is exactly analogous to the one between total orders and partial orders.

Definition 2 (stratified extension [19]). *Let $S = (X, \prec, \sqsubset)$ be a so-structure. A stratified order $\lhd$ on X is a* stratified extension *of S if and only if $(X, \prec, \sqsubset) \subseteq (X, \lhd, \lhd^{\frown})$.*
The set of all stratified extensions of S is denoted as $ext(S)$. ■

Szpilrajn's Theorem [30] states that every poset can be reconstructed by taking the intersection of all of its total order extensions. Janicki and Koutny showed that a similar result holds for so-structures and stratified extensions:

Theorem 1 ([19]). *Let $S = (X, \prec, \sqsubset)$ be a so-structure. Then*
$$S = \left(X, \bigcap_{\lhd \in ext(S)} \lhd, \bigcap_{\lhd \in ext(S)} \lhd^{\frown} \right).$$
□

Using this theorem, we can show the following properties relating so-structures and their stratified extensions.

Corollary 1. *For every so-structure $S = (X, \prec, \sqsubset)$,*

1. $\left(\exists \lhd \in ext(S), \ \alpha \lhd \beta \right) \wedge \left(\exists \lhd \in ext(S), \ \beta \lhd \alpha \right) \implies \left(\exists \lhd \in ext(S), \ \beta \frown_{\lhd} \alpha \right).$
2. $\left(\forall \lhd \in ext(S), \ \alpha \lhd \beta \vee \beta \lhd \alpha \right) \iff \alpha \prec \beta \vee \beta \prec \alpha.$

Proof. **1.** See [19, Theorem 3.6]. **2.** Follows from **1.** and Theorem 1. □

3.2 Combined Traces

Comtraces were introduced in [18] as a generalization of traces to represent so-structures. The *comtrace congruence* is defined via two relations *simultaneity* and *serializability*.

Definition 3 (comtrace alphabet [18]). *Let E be a finite set (of events) and let ser $\subseteq$ sim $\subset E \times E$ be two relations called* serializability *and* simultaneity *respectively and the relation sim is irreflexive and symmetric. The triple $\theta = (E, sim, ser)$ is called a* comtrace alphabet. ■

Intuitively, if $(a,b) \in sim$ then a and b can occur simultaneously (or be a part of a *synchronous* occurrence in the sense of [24]), while $(a,b) \in ser$ means that a and b may occur simultaneously or a may occur before b. We define $\mathbb{S}_\theta$, the set of all possible *steps*, to be the set of all cliques of the graph (E, sim), i.e.,

$$\mathbb{S}_\theta \triangleq \{ A \mid A \neq \emptyset \wedge \forall a, b \in A, \ (a = b \vee (a,b) \in sim) \}.$$

Definition 4 (comtrace congruence [18]). *For a comtrace alphabet $\theta = (E, sim, ser)$, we define $\approx_\theta \subseteq \mathbb{S}_\theta^* \times \mathbb{S}_\theta^*$ to be the relation comprising all pairs (t,u) of step sequences such that*

$$t = wAz \quad and \quad u = wBCz,$$

where $w, z \in \mathbb{S}_\theta^*$ and A, B, C are steps satisfying $B \cup C = A$ and $B \times C \subseteq ser$.

We define comtrace congruence $\equiv_\theta \triangleq \left(\approx_\theta \cup \approx_\theta^{-1} \right)^*$. *We define the comtrace concatenation operator* $\_ \circledast \_$ *as* $[r] \circledast [t] \triangleq [r * t]$. *The quotient monoid $(\mathbb{S}^*/\equiv_\theta, \circledast, [\varepsilon])$ is called the monoid of* comtraces *over θ.* ∎

Note that since *ser* is irreflexive, $B \times C \subseteq ser$ implies that $B \cap C = \emptyset$. We will omit the subscript θ from the comtrace congruence $\approx_\theta$, and write $\equiv$ and $\approx$ when it causes no ambiguity. To shorten our notations, we often write $[s]_\theta$ or $[s]$ instead of $[s]_{\equiv_\theta}$ to denote the comtrace generated by the step sequence s over θ.

Example 1. Let $E = \{a, b, c\}$ where a, b and c are three atomic operations, where

$$a: \ y \leftarrow x + y \qquad b: \ x \leftarrow y + 2 \qquad c: \ y \leftarrow y + 1$$

Assume simultaneous reading is allowed. Then only b and c can be performed simultaneously, and the simultaneous execution of b and c gives the same outcome as executing b followed by c. We can then define the comtrace alphabet $\theta = (E, sim, ser)$, where $sim = \{\{b, c\}\}$ and $ser = \{(b,c)\}$. This yields $\mathbb{S}_\theta = \{\{a\}, \{b\}, \{c\}, \{b,c\}\}$. Thus, $\mathbf{t} = [\{a\}\{b,c\}] = \{\{a\}\{b,c\}, \{a\}\{b\}\{c\}\}$ is a comtrace. But $\{a\}\{c\}\{b\} \notin \mathbf{t}$. ∎

Even though traces are quotient monoids over sequences and comtraces are quotient monoids over step sequences, traces can be regarded as special kinds of comtraces when the relation *ser* = *sim*. For a more detailed discussion on this connection between traces and comtraces, the reader is referred to [22].

Definition 5 ([18]). *Let $u \in \mathbb{S}_\theta^*$. We define the relations $\prec_u, \sqsubset_u \subseteq \Sigma_u \times \Sigma_u$ as:*

1. $\alpha \prec_u \beta \stackrel{df}{\iff} \alpha \vartriangleleft_u \beta \wedge (\ell(\alpha), \ell(\beta)) \notin ser,$
2. $\alpha \sqsubset_u \beta \stackrel{df}{\iff} \alpha \vartriangleleft_u^\frown \beta \wedge (\ell(\beta), \ell(\alpha)) \notin ser.$ ∎

It is worth noting that the structure $(\Sigma_u, \prec_u, \sqsubset_u, \ell)$ is exactly the *cd-graph* (cf. Definition 11) that represents the comtrace $[u]$. This gives us some intuition on how Koutny and Kleijn constructed the cd-graph definition in [26]. We also observe that $(\Sigma_u, \prec_u, \sqsubset_u)$ is usually *not* a so-structure since $\prec_u$ and $\sqsubset_u$ describe only basic "local" causality and weak causality invariants of the event occurrences of u by considering pairwise serializable relationships of event occurrences. Hence, $\prec_u$ and $\sqsubset_u$ might not capture "global" invariants that can be inferred from S2–S4 of Definition 1. To ensure all invariants are included, we need the following $\Diamond$-closure operator.

Definition 6 ([18]). *For every relational structure $S = (X, R_1, R_2)$ we define $S^\Diamond$ as*
$$S^\Diamond \triangleq \left(X, (R_1 \cup R_2)^* \circ R_1 \circ (R_1 \cup R_2)^*, (R_1 \cup R_2)^* \setminus id_X \right). \qquad \blacksquare$$

Intuitively $\Diamond$-closure is a generalization of transitive closure for relations to relational structures. The motivation is that for appropriate relations R_1 and R_2 (see assertion (3) of Proposition 2), the relational structure $(X, R_1, R_2)^\Diamond$ is a so-structure. The $\Diamond$-closure operator satisfies the following properties:

Proposition 2 ([18]). *Let $S = (X, R_1, R_2)$ be a relational structure.*

1. *If R_2 is irreflexive then $S \subseteq S^\Diamond$.*
2. *$(S^\Diamond)^\Diamond = S^\Diamond$.*
3. *$S^\Diamond$ is a so-structure if and only if $(R_1 \cup R_2)^* \circ R_1 \circ (R_1 \cup R_2)^*$ is irreflexive.*
4. *If S is a so-structure then $S = S^\Diamond$.*
5. *If S be a so-structure and $S_0 \subseteq S$, then $S_0^\Diamond \subseteq S$ and $S_0^\Diamond$ is a so-structure.* $\qquad \square$

Definition 7. *Given a step sequence $u \in \mathbb{S}_\theta^*$ and its respective comtrace $\mathbf{t} = [u] \in \mathbb{S}_\theta^* / \equiv$, we define the relational structures $S_\mathbf{t}$ as:*
$$S_\mathbf{t} = \left(\Sigma_\mathbf{t}, \prec_\mathbf{t}, \sqsubset_\mathbf{t} \right) \triangleq \left(\Sigma_u, \prec_u, \sqsubset_u \right)^\Diamond. \qquad \blacksquare$$

The relational structure $S_\mathbf{t}$ is called the *so-structure defined by the comtrace* $\mathbf{t} = [u]$, where $\Sigma_\mathbf{t}$, $\prec_\mathbf{t}$ and $\sqsubset_\mathbf{t}$ are used to denote the event occurrence set, causality relation and weak causality relation induced by the comtrace $\mathbf{t}$ respectively. The following nontrivial theorem and its corollary justifies the name by showing that step sequences in a comtrace $\mathbf{t}$ are exactly stratified extension of the so-structure $S_\mathbf{t}$, and that $S_\mathbf{t}$ is uniquely defined for the comtrace $\mathbf{t}$ regardless of the choice of $u \in \mathbf{t}$.

Theorem 2 ([18]). *For each $\mathbf{t} \in E^* / \equiv_\theta$, the relational structure $S_\mathbf{t}$ is a so-structure and $ext(S_\mathbf{t}) = \left\{ \lhd_u \mid u \in \mathbf{t} \right\}$.* $\qquad \square$

Corollary 2. *For all $\mathbf{t}, \mathbf{q} \in E^* / \equiv_\theta$,*

1. *$\mathbf{t} = \mathbf{q} \implies S_\mathbf{t} = S_\mathbf{q}$*
2. *$S_\mathbf{t} = \left(\Sigma_\mathbf{t}, \prec_\mathbf{t}, \sqsubset_\mathbf{t} \right) = \left(\Sigma_\mathbf{t}, \bigcap_{w \in \mathbf{t}} \lhd_w, \bigcap_{w \in \mathbf{t}} \widehat{\lhd_w} \right)$* $\qquad \square$

4 Comtraces as Labeled Stratified Order Structures

Even though Theorem 2 shows that each comtrace can be represented uniquely by a labeled so-structure, it does not give us an explicit definition of how these labeled so-structures look like. In this section, we will give an exact definition of labeled so-structures that represent comtraces. To provide us with more intuition, we first recall how Mazurkiewicz traces can be characterized as labeled posets.

A *trace concurrent alphabet* is a pair (E, ind), where ind is a symmetric irreflexive binary relation on the finite set E. A *trace congruence* $\equiv_{ind}$ can then be defined as the smallest equivalence relation such that for all sequences $uabv, ubav \in E^*$, if $(a, b) \in ind$, then $uabv \equiv_{ind} ubav$. The elements of $E^* / \equiv_{ind}$ are called *traces*.

Traces can also be defined alternatively as posets whose elements are labeled with symbols of a concurrent alphabet (E, ind) satisfying certain conditions.

Given a binary relation $R \subseteq X$, the *covering relation* of R is defined as $R^{\text{cov}} \triangleq \{(x, y) \mid x\, R\, y \wedge \neg \exists z,\, x\, R\, z\, R\, y\}$. An alternative definition of Mazurkiewicz trace is:

Definition 8 (cf. [31]). *A trace over a concurrent alphabet (E, ind) is a finite labeled poset $(X, \prec, \lambda)$, where $\lambda : X \to E$ is a labeling function, such that for all $\alpha, \beta \in X$,*

1. *$\alpha \prec^{\text{cov}} \beta \implies (\lambda(\alpha), \lambda(\beta)) \notin ind$, and*
2. *$(\lambda(\alpha), \lambda(\beta)) \notin ind \implies \alpha \prec \beta \vee \beta \prec \alpha$.* ∎

A trace in this definition is only identified unique up to *label-preserving isomorphism*. The first condition says that immediately causally related event occurrences must be labeled with dependent events. The second condition ensures that any two event occurrences with dependent labels must be causally related. The first condition is particularly important since two immediately causally related event occurrences will occur next to each other in at least one of its linear extensions. This is the key to relate Definition 8 with quotient monoid definition of traces. Thus, we would like to establish a similar relationship for comtraces. An immediate technical difficulty is that weak causality might be cyclic, so the notion of "immediate weak causality" does not make sense. However, we can still deal with cycles of a so-structure by taking advantage of the following simple fact: *the weak causality relation is a strict preorder*.

Let $S = (X, \prec, \sqsubset)$ be a so-structure. We define the relation $\equiv_{\sqsubset} \subseteq X \times X$ as

$$\alpha \equiv_{\sqsubset} \beta \overset{df}{\iff} \alpha = \beta \vee (\alpha \sqsubset \beta \wedge \beta \sqsubset \alpha)$$

Since $\sqsubset$ is a strict preorder, it follows that $\equiv_{\sqsubset}$ is an equivalence relation. The relation $\equiv_{\sqsubset}$ will be called the $\sqsubset$-*cycle equivalence relation* and an element of the quotient set $X / \equiv_{\sqsubset}$ will be called a $\sqsubset$-*cycle equivalence class*. We then define the following binary relations $\widehat{\prec}$ and $\widehat{\sqsubset}$ on the quotient set $X / \equiv_{\sqsubset}$ as

$$[\alpha] \widehat{\prec} [\beta] \overset{df}{\iff} ([\alpha] \times [\beta]) \cap \prec \neq \emptyset \quad \text{and} \quad [\alpha] \widehat{\sqsubset} [\beta] \overset{df}{\iff} ([\alpha] \times [\beta]) \cap \sqsubset \neq \emptyset \quad (4.1)$$

Using this quotient construction, every so-structure, whose weak causality relation might be cyclic, can be uniquely represented by an *acyclic* quotient so-structure.

Proposition 3. *The relational structure $S / \equiv_{\sqsubset} \triangleq (X / \equiv_{\sqsubset}, \widehat{\prec}, \widehat{\sqsubset})$ is a so-structure, the relation $\widehat{\sqsubset}$ is a partial order, and for all $x, y \in X$,*

1. *$\alpha \prec \beta \iff [\alpha] \widehat{\prec} [\beta]$*
2. *$\alpha \sqsubset \beta \iff [\alpha] \widehat{\sqsubset} [\beta] \vee (\alpha \neq \beta \wedge [\alpha] = [\beta])$*

Proof. Follows from Definition 1. □

Using (4.1) and Theorem 1, it is not hard to prove the following simple yet useful properties of $\sqsubset$-cycle equivalence classes.

Proposition 4. *Let $S = (X, \prec, \sqsubset)$ be a so-structure. We use u and v to denote some step sequences over $\wp^{\backslash \{\emptyset\}}(X)$. Then for all $\alpha, \beta \in X$,*

1. $[\alpha] = [\beta] \iff \forall \lhd \in ext(S), \; \alpha \simeq_{\lhd} \beta$
2. $\exists \lhd \in ext(S), \; \Omega_{\lhd} = u[\alpha]v$
3. $[\alpha]\hat{\sqsubset}^{\mathsf{cov}}[\beta] \implies \exists \lhd \in ext(S), \; \Omega_{\lhd} = u[\alpha][\beta]v$ $\square$

Each $\sqsubset$-cycle equivalence class is what Juhás, Lorenz and Mauser called a *synchronous step* [24,23]. They also used equivalence classes to capture synchronous steps but only for the special class of *synchronous closed* so-structures, where $(\sqsubset \setminus \prec) \cup id_X$ is an equivalence relation. We extend their ideas by using $\sqsubset$-cycle equivalence classes to capture what we will call *non-serializable sets* in arbitrary so-structures. The name is justified in assertion (1) of Proposition 4 stating that two elements belong to the same non-serializable set of a so-structure S iff they must be executed simultaneously in every stratified extension of S. Furthermore, we show in assertion (2) that all elements of a non-serializable set must occur together as a single step in at least one stratified extension of S. Assertion (3) gives a sufficient condition for two non-serializable sets to occur as consecutive steps in at least one stratified extension of S.

Before we proceed to define comtrace using labeled so-structure, we need to define *label-preserving isomorphisms* for labeled so-structures more formally. A tuple $T = (X,P,Q,\lambda)$ is a *labeled relational structure* iff (X,P,Q) is a relational structure and λ is a function with domain X. If (X,P,Q) is a so-structure, then T is a *labeled so-structure*.

Definition 9 (label-preserving isomorphism). *Given two labeled relational structures* $T_1 = (X_1,P_1,Q_1,\lambda_1)$ *and* $T_2 = (X_2,P_2,Q_2,\lambda_2)$*, we write* $T_1 \cong T_2$ *to denote that* T_1 *and* T_2 *are* label-preserving isomorphic *(lp-isomorphic). In other words, there is a bijection* $f : X_1 \to X_2$ *such that for all* $\alpha,\beta \in X_1$,

1. $(\alpha,\beta) \in P_1 \iff (f(\alpha),f(\beta)) \in P_2$
2. $(\alpha,\beta) \in Q_1 \iff (f(\alpha),f(\beta)) \in Q_2$
3. $\lambda_1(\alpha) = \lambda_2(f(\alpha))$

Such function f is called a label-preserving isomorphism *(lp-isomorphism).*   ∎

Note that all notations, definitions and results for so-structures are applicable to labeled so-structures. We also write $[T]$ or $[X,P,Q,\lambda]$ to denote the lp-isomorphic class of a labeled relational structure $T = (X,P,Q,\lambda)$. We will not distinguish an lp-isomorphic class $[T]$ with a single labeled relational structure T when it does not cause ambiguity.

We are now ready to give an alternative definition for comtraces. To avoid confusion with the comtrace notion by Janicki and Koutny in [18], we will use the term *lsos-comtrace* to denote a comtrace defined using our definition.

Definition 10 (lsos-comtrace). *Given a comtrace alphabet* $\theta = (E, sim, ser)$*, a* lsos-comtrace *over θ is (an lp-isomorphic class of) a finite labeled so-structure* $[X, \prec, \sqsubset, \lambda]$ *such that $\lambda : X \to E$ and for all* $\alpha,\beta \in X$,

LC1: $[\alpha](\hat{\sqsubset}^{\mathsf{cov}} \cap \hat{\frown})[\beta] \implies \lambda([\alpha]) \times \lambda([\beta]) \not\subseteq ser$

LC2: $[\alpha](\hat{\sqsubset}^{\mathsf{cov}} \setminus \hat{\frown})[\beta] \implies \lambda([\beta]) \times \lambda([\alpha]) \not\subseteq ser$

LC3: $\forall A,B \in \wp^{\setminus\{\emptyset\}}([\alpha]), \; A \cup B = [\alpha] \implies \lambda(A) \times \lambda(B) \not\subseteq ser$

LC4: $(\lambda(\alpha),\lambda(\beta)) \notin ser \implies \alpha \prec \beta \vee \beta \sqsubset \alpha$

LC5: $(\lambda(\alpha), \lambda(\beta)) \notin sim \implies \alpha \prec \beta \vee \beta \prec \alpha$

We write $\mathsf{LCT}(\theta)$ *to denote the class of all lsos-comtraces over* θ. ■

Example 2. Let $E = \{a,b,c\}$, $sim = \{\{a,b\},\{a,c\},\{b,c\}\}$ and $ser = \{(a,b),(b,a),$ $(a,c)\}$. Then we have $\mathbb{S} = \{\{a\},\{b\},\{c\},\{b,c\}\}$. The lp-isomorphic class of the labeled so-structure $T = (X,\prec,\sqsubset,\lambda)$ depicted in Figure 1 (the dotted edges denote $\sqsubset$ relation and the solid edges denote both $\prec$ and $\sqsubset$ relations) is a lsos-comtrace. The graph in Figure 2 represents the labeled quotient so-structure $T/\!\!\equiv_\sqsubset = (X/\!\!\equiv_\sqsubset, \widehat{\prec}, \widehat{\sqsubset}, \lambda')$ of T, where we define $\lambda'(A) = \{\lambda(x) \mid x \in A\}$.
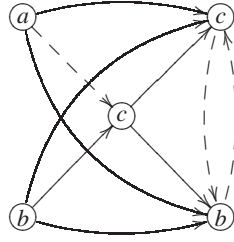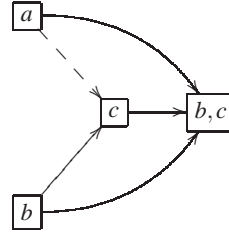

Fig. 1: lsos-comtrace $[T]$ Fig. 2: the quotient structure $T/\!\!\equiv_\sqsubset$ of T

The lsos-comtrace $[T]$ actually corresponds to the comtrace $[\{a,b\}\{c\}\{b,c\}]$, and we will show this relationship formally in Section 5. ■

Remark 1. Definition 10 can be extended to define *infinite comtrace* as follows. Instead of asking X to be finite, we require a labeled so-structure to be *initially finite* (cf. [19]), i.e., $\{\alpha \in X \mid \alpha \sqsubset \beta\}$ is finite for all $\beta \in X$. The initially-finiteness not only gives us a sensible interpretation that every event only causally depends on finitely many events, but also guarantees that the covering relations of $\widehat{\prec}$ and $\widehat{\sqsubset}$ are well-defined. ■

Since each lsos-comtrace is defined as a class of lp-isomorphic labeled so-structures, dealing with lsos-comtrace might seem tricky. Fortunately, the *no autoconcurrency* property, i.e., the relation *ser* is irreflexive, gives us a *canonical* way to enumerate the events of a lsos-comtrace very similar to how the events of a comtrace are enumerated.

Given a step sequence $s = A_1 \ldots A_k$ and any function f defined on $\bigcup_{i=1}^k A_i$, we define $\mathsf{map}(f,s) \triangleq f(A_1) \ldots f(A_k)$, i.e., the step sequence derived from s by applying the function f successively on each A_i. Note that $f(A_i)$ denotes the *image* of A_i under f.

Given a lsos-comtrace $T = [X,\prec,\sqsubset,\lambda]$ over a comtrace alphabet $\theta = (E,sim,ser)$, a stratified order $\lhd \in ext(T)$ can be seen as a step sequence $\Omega_\lhd = A_1 \ldots A_k$.

Proposition 5.   *1. For every i ($1 \leq i \leq k$), $|A_i| = |\lambda(A_i)|$*
2. $\mathsf{map}(\lambda, \Omega_\lhd) = \lambda(A_1) \ldots \lambda(A_k) \in \mathbb{S}_\theta^*.$ □

Proposition 5 ensures that $u = \mathsf{map}(\lambda, \Omega_\lhd)$ is a valid step sequence over θ. Recall that $\overline{u} = \overline{A_1} \ldots \overline{A_k}$ denotes the enumerated step sequence of u and Σ_u denotes the set of

event occurrences. Define a *bijection* $\xi_u : \Sigma_u \to X$ as

$$\xi_u(\alpha) = x \overset{df}{\Longleftrightarrow} \alpha \in \overline{A_i} \wedge x \in A_i \wedge \lambda(x) = \ell(\alpha)$$

By Proposition 5, the function ξ_u is well-defined. Moreover, we can show that ξ_u is uniquely determined by T regardless of the choice of $\lhd \in ext(T)$.

Proposition 6. *Given $\lhd_1, \lhd_2 \in ext(T)$, let $v = \mathsf{map}(\lambda, \Omega_{\lhd_1})$ and $w = \mathsf{map}(\lambda, \Omega_{\lhd_2})$. Then $\xi_v = \xi_w$.* $\qquad\square$

Henceforth, we will ignore subscripts and reserve the notation ξ to denote the kind of mappings as defined above. We then define the *enumerated so-structure* of T to be the labeled so-structure $T_0 = (\Sigma, \prec_0, \sqsubset_0, \ell)$, where $\Sigma = \Sigma_u$ for $u = \mathsf{map}(\lambda, \Omega_{\lhd})$ and $\lhd \in ext(T)$; and the relations $\prec_0, \sqsubset_0 \subseteq \Sigma \times \Sigma$ are defined as

$$\alpha \prec_0 \beta \overset{df}{\Longleftrightarrow} \xi(\alpha) \prec \xi(\beta) \quad \text{and} \quad \alpha \sqsubset_0 \beta \overset{df}{\Longleftrightarrow} \xi(\alpha) \sqsubset \xi(\beta)$$

Clearly, the enumerated so-structure T_0 can be uniquely determined from T using the preceding definition. From our construction, we can easily show the following important relationships:

Proposition 7.   *1. T_0 and T are lp-isomorphic under the mapping ξ.*
  2. The labeled so-structures $(\Sigma, \lhd_u, \lhd_u^\frown, \ell)$ and $(X, \lhd, \lhd^\frown, \lambda)$ are lp-isomorphic under the mapping ξ and $\lhd_u \in ext(T_0)$. $\qquad\square$

In other words, the mapping $\xi : \Sigma \to X$ plays the role of both the lp-isomorphism from T_0 to T and the lp-isomorphism from the stratified extension $(\Sigma, \lhd_u)$ of T_0 to the stratified extension $(X, \lhd)$ of T. These relationships can be best captured using the commutative diagram on the right.

$$
\begin{array}{ccc}
(\Sigma, \prec_0, \sqsubset_0, \ell) & \overset{\xi}{\longrightarrow} & (X, \prec, \sqsubset, \lambda) \\
{\scriptstyle id_\Sigma}\big\downarrow & & \big\downarrow{\scriptstyle id_X} \\
(\Sigma, \lhd_u, \lhd_u^\frown, \ell) & \overset{\xi}{\longrightarrow} & (X, \lhd, \lhd^\frown, \lambda)
\end{array}
$$

We can even observe further that two lsos-comtraces are identical if and only if they define the same enumerated so-structure. Henceforth, we will call an enumerated so-structure defined by a lsos-comtrace T *the canonical representation* of T.

Recently, inspired by the dependency graph notion for Mazurkiewicz traces (cf. [5, Chapter 2]), Kleijn and Koutny claimed without proof that their *combined dependency graph* notion is another alternative way to define comtraces [26]. In Section 5, we will give a detailed proof of their claim.

Definition 11 (combined dependency graph [26]). *Given an comtrace alphabet $\theta = (E, ser, sim)$, a* combined dependency graph *(cd-graph) over θ is (a lp-isomorphic class of) a finite labeled relational structure $D = [X, \longrightarrow, \dashrightarrow, \lambda]$ such that $\lambda : X \to E$, the relations $\longrightarrow, \dashrightarrow$ are irreflexive, $D^\lozenge$ is a so-structure, and for all $\alpha, \beta \in X$,*

  CD1: $(\lambda(\alpha), \lambda(\beta)) \notin sim \implies \alpha \longrightarrow \beta \vee \beta \longrightarrow \alpha$
  CD2: $(\lambda(\alpha), \lambda(\beta)) \notin ser \implies \alpha \longrightarrow \beta \vee \beta \dashrightarrow \alpha$
  CD3: $\alpha \longrightarrow \beta \implies (\lambda(\alpha), \lambda(\beta)) \notin ser$
  CD4: $\alpha \dashrightarrow \beta \implies (\lambda(\beta), \lambda(\alpha)) \notin ser$

We will write $\mathsf{CDG}(\theta)$ to denote the class of all cd-graphs over θ. $\qquad\blacksquare$

Cd-graphs can be seen as reduced graph-theoretic representations for lsos-comtraces, where some arcs that can be recovered using $\Diamond$-closure are omitted. It is interesting to observe that the non-serializable sets of a cd-graph are exactly the *strongly connected components* of the directed graph $(X, \dashrightarrow)$ and can easily be found in time $O(|X| + |\dashrightarrow|)$ using any standard algorithm (cf. [2, Section 22.5]).

Remark 2. Cd-graphs were called *dependence comdags* in [26]. But this name could be misleading since the directed graph $(X, \dashrightarrow)$ is not necessarily acyclic. For example, the graph on the right is the cd-graph that corresponds to the lsos-comtrace from Figure 1, but it is not acyclic. (Here, we use the dotted edges to denote $\dashrightarrow$ and the solid edges to denote *only* $\longrightarrow$.) Thus, we use the name "combined dependency graph" instead. ∎
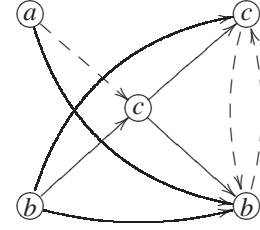
5 Representation Theorems

This section contains the main technical contribution of this paper by showing that for a given comtrace alphabet θ, $\mathbb{S}^*/\equiv_\theta$, $\mathsf{LCT}(\theta)$ and $\mathsf{CDG}(\theta)$ are three equivalent ways of talking about the same class of objects. We will next prove the first representation theorem which establishes the representation mappings between $\mathbb{S}^*/\equiv_\theta$ and $\mathsf{LCT}(\theta)$.

5.1 Representation Theorem for Comtraces and lsos-Comtraces

Proposition 8. *Let $S_0 = (X, \prec_0, \sqsubset_0)$ and $S_1 = (X, \prec_1, \sqsubset_1)$ be stratified order structures such that $ext(S_0) \subseteq ext(S_1)$. Then $S_1 \subseteq S_0$.*

Proof. Follows from Theorem 1. □

For the next two lemmata, we let T be a lsos-comtrace over a comtrace alphabet $\theta = (E, sim, ser)$. Let $T_0 = (\Sigma_u, \prec_0, \sqsubset_0, \ell)$ be the canonical representation of T. Let $\lhd_0 \in ext(T_0)$ and $u = \mathsf{map}(\ell, \Omega_{\lhd_0})$. Since u is a valid step sequence in $\mathbb{S}^*$ (by Proposition 5), we can construct $S_{[u]} = (\Sigma_u, \prec_{[u]}, \sqsubset_{[u]})$ from Definition 7. Our goal is to show that the stratified order $S_{[u]}$ defined by the comtrace $[u]$ is exactly $(\Sigma_u, \prec_0, \sqsubset_0)$.

Lemma 1. $S_{[u]} \subseteq (\Sigma_u, \prec_0, \sqsubset_0)$.

Proof. By Proposition 2, to show $S_{[u]} = (\Sigma_u, \prec_u, \sqsubset_u)^\Diamond \subseteq (\Sigma_u, \prec_0, \sqsubset_0)$, it suffices to show that $(\Sigma_u, \prec_u, \sqsubset_u) \subseteq (\Sigma_u, \prec_0, \sqsubset_0)$. Since T_0 is the canonical representation of T, it is important to observe that $\lhd_0 = \lhd_u$.

($\prec_u \subseteq \prec_0$): Assume $\alpha \prec_u \beta$. Then from Definition 5, $\alpha \lhd_u \beta \wedge (\ell(\alpha), \ell(\beta)) \notin ser$. Since $(\ell(\alpha), \ell(\beta)) \notin ser$, it follows from Definition 10 that $\alpha \prec_0 \beta$ or $\beta \sqsubset_0 \alpha$. Suppose for a contradiction that $\beta \sqsubset_0 \alpha$, then by Theorem 1, $\forall \lhd \in ext(T_0)$, $\beta \lhd^\frown \alpha$. But since we assume that $\lhd_0 \in ext(T_0)$, it follows that $\lhd_u \in ext(T_0)$ and $\alpha \lhd_u \beta$, a contradiction. Hence, we have shown $\alpha \prec_0 \beta$.

($\sqsubset_u \subseteq \sqsubset_0$): Can be shown in a similar way. □

Lemma 2. $S_{[u]} \supseteq (\Sigma_u, \prec_0, \sqsubset_0)$.

In this proof, we will include subscripts for equivalence classes to avoid confusing the elements from quotient set $\Sigma_u/\equiv_{\sqsubset_0}$ with the elements from the quotient comtrace monoid $\mathbb{S}^*/\equiv_\theta$. In other words, we write $[\alpha]_{\equiv_{\sqsubset_0}}$ to denote an element of the quotient set $\Sigma_u/\equiv_{\sqsubset_0}$, and write $[u]_\theta$ to denote the comtrace generated by u.

Proof (of Lemma 2). Let $S' = (\Sigma_u, \prec_0, \sqsubset_0)$. To show $S_{[u]} \supseteq S'$, by Proposition 8, it suffices to show $ext(S_{[u]}) \subseteq ext(S')$. From Theorem 2, we know that $ext(S_{[u]_\theta}) = \{\lhd_w \mid w \in [u]_\theta\}$. Thus we only need to show that for all $w \in [u]_\theta$, $\lhd_w \in ext(S')$.

We observe that from u, by Definition 4, we can generate all the step sequences in the comtrace $[u]_\theta$ in stages using the following recursive definition:

$$D^0(u) \triangleq \{u\}$$
$$D^n(u) \triangleq \{w \mid w \in D^{n-1}(u) \vee \exists v \in D^{n-1}(u),\ (\ v \approx_\theta w \vee v \approx_\theta^{-1} w)\}$$

Since the set $[u]_\theta$ is finite, $[u]_\theta = D^n(u)$ for some stage $n \geq 0$. For the rest of the proof, we will prove by induction on n that for all $n \in \mathbb{N}$, if $w \in D^n(u)$ then $\lhd_w \in ext(S)$.
Base case: When $n = 0$, $D^0(u) = \{u\}$. Since $\lhd_0 \in ext(T)$, it follows from Proposition 7 that $\lhd_u \in ext(S')$.
Inductive case: When $n > 0$, let w be an element of $D^n(u)$. Then either $w \in D^{n-1}(u)$ or $w \in (D^n(u) \setminus D^{n-1}(u))$. For the former case, by inductive hypothesis, $\lhd_w \in ext(S')$. For the latter case, there must be some element $v \in D^{n-1}(u)$ such that $v \approx_\theta w$ or $v \approx_\theta^{-1} w$. By induction hypothesis, we already known $\lhd_v \in ext(S')$. We want to show that $\lhd_w \in ext(S')$. There are two cases to consider:

Case (i):
When $v \approx_\theta w$, by Definition 4, there are some $y, z \in E_\theta^*$ and steps $A, B, C \in \mathbb{S}$ such that $v = yAz$ and $w = yBCz$ where A, B, C satisfy $B \cap C = \emptyset$ and $B \cup C = A$ and $B \times C \subseteq ser$. Let $\overline{v} = \overline{y}\,\overline{A}\,\overline{z}$ and $\overline{w} = \overline{y}\,\overline{B}\,\overline{C}\,\overline{z}$ be enumerated step sequences of v and w respectively.

Suppose for a contradiction that $\lhd_w \notin ext(S')$. By Definition 2, there are $\alpha \in \overline{C}$ and $\beta \in \overline{B}$ such that $\alpha \sqsubset_0 \beta$. We now consider the quotient set $\overline{A}/\equiv_{\sqsubset_0}$. By Proposition 4 (1), $\overline{A}/\equiv_{\sqsubset_0} \subseteq \Sigma_u/\equiv_{\sqsubset_0}$. Since $\alpha \sqsubset_0 \beta$, it follows that $[\alpha]_{\equiv_{\sqsubset_0}} \hat{\sqsubset}_0 [\beta]_{\equiv_{\sqsubset_0}}$. Thus, from the fact that $\hat{\sqsubset}_0$ is partial order, there must exists a chain

$$[\alpha]_{\equiv_{\sqsubset_0}} = [\gamma_1]_{\equiv_{\sqsubset_0}} \hat{\sqsubset}_0^{\text{cov}} [\gamma_2]_{\equiv_{\sqsubset_0}} \hat{\sqsubset}_0^{\text{cov}} \ldots \hat{\sqsubset}_0^{\text{cov}} [\gamma_k]_{\equiv_{\sqsubset_0}} = [\beta]_{\equiv_{\sqsubset_0}} \tag{5.1}$$

Then by Theorem 1 and the fact that $\lhd_v \in ext(S')$, we know that $\gamma_i \in \overline{A}$ for all i. In other words, since the chain (5.1) implies that every γ_i must always occur between α and β in all stratified extensions of S' and $\alpha, \beta \in \overline{A}$, we also have $\gamma_i \in \overline{A}$. Hence, by Proposition 4 (1), we have $[\gamma_i]_{\equiv_{\sqsubset_0}} \subseteq \overline{A}$ for all i, $1 \leq i \leq k$. Also from LC3 of Definition 10 and that $B \times C \subseteq ser$, we know that for each γ_i, either $[\gamma_i]_{\equiv_{\sqsubset_0}} \subseteq \overline{B}$ or $[\gamma_i]_{\equiv_{\sqsubset_0}} \subseteq \overline{C}$. Now we note that the first element on the chain $[\gamma_1]_{\equiv_{\sqsubset_0}} = [\alpha]_{\equiv_{\sqsubset_0}} \subseteq \overline{C}$ and the last element on the chain $[\gamma_k]_{\equiv_{\sqsubset_0}} = [\beta]_{\equiv_{\sqsubset_0}} \subseteq \overline{B}$. Thus, there exist two consecutive elements $[\gamma_i]_{\equiv_{\sqsubset_0}}$ and $[\gamma_{i+1}]_{\equiv_{\sqsubset_0}}$ on the chain such that $[\gamma_i]_{\equiv_{\sqsubset_0}} \subseteq \overline{C}$ and $[\gamma_{i+1}]_{\equiv_{\sqsubset_0}} \subseteq \overline{B}$. But then it follows that

(a) $[\gamma_{i+1}]_{\equiv_{\sqsubset_0}} \times [\gamma_i]_{\equiv_{\sqsubset_0}} \subseteq ser$ and $[\gamma_i]_{\equiv_{\sqsubset_0}} \hat{\sqsubset}_0^{\text{cov}} [\gamma_{i+1}]_{\equiv_{\sqsubset_0}}$

(b) $\neg\big([\gamma_i]_{\equiv_{\sqsubset_0}} \overset{\curvearrowright}{\sim}_0 [\gamma_{i+1}]_{\equiv_{\sqsubset_0}}\big)$ since $\lhd_v \in ext(S')$ and $\gamma_i \frown_{\lhd_v} \gamma_{i+1}$

These contradict $\mathsf{LC2}$ of Definition 10 since T_0 is a lsos-comtrace.

Case (ii):
When $v \approx_\theta^{-1} w$, by Definition 4, there are some $y, z \in E_\theta^*$ and steps $A, B, C \in \mathbb{S}$ such that $v = yBCz$ and $w = yAz$ where A, B, C satisfy $B \cap C = \emptyset$ and $B \cup C = A$ and $B \times C \subseteq ser$. Let $\overline{v} = \overline{y}\,\overline{B}\,\overline{C}\,\overline{z}$ and $\overline{w} = \overline{y}\,\overline{A}\,\overline{z}$ be enumerated step sequences of v and w respectively.

Suppose for a contradiction that $\lhd_w \notin ext(S')$. By Definition 2, there are $\alpha \in \overline{B}$ and $\beta \in \overline{C}$ such that $\alpha \prec_0 \beta$. By Proposition 4 (1), $\overline{A}/{\equiv_{\sqsubset_0}} \subseteq \Sigma_u/{\equiv_{\sqsubset_0}}$. Thus, using a dual argument to the proof of Case (i), we can build a chain

$$[\alpha]_{\equiv_{\sqsubset_0}} = [\gamma_1]_{\equiv_{\sqsubset_0}} \overset{\curvearrowright\mathsf{cov}}{\sqsubset_0} [\gamma_2]_{\equiv_{\sqsubset_0}} \overset{\curvearrowright\mathsf{cov}}{\sqsubset_0} \cdots \overset{\curvearrowright\mathsf{cov}}{\sqsubset_0} [\gamma_k]_{\equiv_{\sqsubset_0}} = [\beta]_{\equiv_{\sqsubset_0}} \tag{5.2}$$

We then argue that there are two consecutive elements on the chain such that $[\gamma_i]_{\equiv_{\sqsubset_0}} \subseteq \overline{B}$ and $[\gamma_{i+1}]_{\equiv_{\sqsubset_0}} \subseteq \overline{C}$, which implies

(a) $[\gamma_i]_{\equiv_{\sqsubset_0}} \times [\gamma_{i+1}]_{\equiv_{\sqsubset_0}} \subseteq ser$ and $[\gamma_i]_{\equiv_{\sqsubset_0}} \overset{\curvearrowright\mathsf{cov}}{\sqsubset_0} [\gamma_{i+1}]_{\equiv_{\sqsubset_0}}$
(b) $[\gamma_i]_{\equiv_{\sqsubset_0}} \overset{\curvearrowright}{\sim}_0 [\gamma_{i+1}]_{\equiv_{\sqsubset_0}}$ since $\lhd_v \in ext(S')$ and $\gamma_i \lhd_v \gamma_{i+1}$

These contradict $\mathsf{LC1}$ of Definition 10. $\qquad\square$

We also need to show that the labeled so-structure defined from each comtrace is indeed a lsos-comtrace. In other words, we need to show the following lemma.

Lemma 3. *Let $\theta = (E, sim, ser)$ be a comtrace alphabet. Given a step sequence $u \in \mathbb{S}_\theta^*$, the lp-isomorphic class $\big[\Sigma_{[u]}, \prec_{[u]}, \sqsubset_{[u]}, \ell\big]$ is a lsos-comtrace over θ.* $\qquad\square$

The proof of this lemma is straightforward by checking that $\big[\Sigma_{[u]}, \prec_{[u]}, \sqsubset_{[u]}, \ell\big]$ satisfies all conditions $\mathsf{LC1}$–$\mathsf{LC5}$.

Definition 12 (representation mappings $\mathsf{ct2lct}$ and $\mathsf{lct2ct}$). *Let θ be a comtrace alphabet.*

1. *The mapping $\mathsf{ct2lct} : \mathbb{S}_\theta^*/{\equiv_\theta} \to \mathsf{LCT}(\theta)$ is defined as*

$$\mathsf{ct2lct}(\mathbf{t}) \triangleq [\Sigma_\mathbf{t}, \prec_\mathbf{t}, \sqsubset_\mathbf{t}, \ell],$$

   where the function $\ell : \Sigma_s \to E$ is defined in Section 2.2 and $S_\mathbf{t} = (\Sigma_\mathbf{t}, \prec_\mathbf{t}, \sqsubset_\mathbf{t})$ is the so-structure defined by the comtrace $\mathbf{t}$ from Definition 7.
2. *The mapping $\mathsf{lct2ct} : \mathsf{LCT}(\theta) \to \mathbb{S}_\theta^*/{\equiv_\theta}$ is defined as*

$$\mathsf{lct2ct}\big((X, \prec, \sqsubset, \lambda)\big) \triangleq \Big\{ map(\lambda, \Omega_\lhd) \mid \lhd \in ext\big((X, \prec, \sqsubset)\big) \Big\}. \qquad\blacksquare$$

Intuitively, the mapping $\mathsf{ct2lct}$ is used to convert a comtrace to lsos-comtrace while the mapping $\mathsf{lct2ct}$ is used to transform a lsos-comtrace into a comtrace. The fact that $\mathsf{ct2lct}$ and $\mathsf{lct2ct}$ are valid representation mappings for $\mathbb{S}_\theta^*/{\equiv_\theta}$ and $\mathsf{LCT}(\theta)$ will be shown in the following theorem.

Theorem 3 (The 1st Representation Theorem). *Let θ be a comtrace alphabet.*

1. *For every* $\mathbf{t} \in \mathbb{S}_\theta^*/\equiv_\theta$, $\mathsf{lct2ct} \circ \mathsf{ct2lct}(\mathbf{t}) = \mathbf{t}$.
2. *For every* $T \in \mathsf{LCT}(\theta)$, $\mathsf{ct2lct} \circ \mathsf{lct2ct}(T) = T$.

Proof. **1.** The fact that $ran(\mathsf{ct2lct}) \subseteq \mathsf{LCT}(\theta)$ follows from Lemma 3. Now for a given $\mathbf{t} \in \mathbb{S}_\theta^*/\equiv_\theta$, we have $\mathsf{ct2lct}(\mathbf{t}) = (\Sigma_\mathbf{t}, \prec_\mathbf{t}, \sqsubset_\mathbf{t}, \ell)$. Thus, it follows that

$$
\begin{aligned}
\mathsf{lct2ct}(\mathsf{ct2lct}(t)) &= \left\{ map(\ell, \Omega_\lhd) \mid \lhd \in ext(S_\mathbf{t}) \right\} \\
&= \left\{ map(\ell, \Omega_\lhd) \mid \lhd \in \{\lhd_s \mid s \in \mathbf{t}\} \right\} \qquad \langle \text{ by Theorem 2 } \rangle \\
&= \left\{ map(\ell, \Omega_{\lhd_s}) \mid s \in \mathbf{t} \right\} = \mathbf{t}
\end{aligned}
$$

2. Assume $T_0 = (\Sigma, \prec_0, \sqsubset_0, \ell)$ is the canonical representation of T. Observe that since $T_0 \cong T$, we have $\left\{ map(\ell, \Omega_\lhd) \mid \lhd \in ext(T_0) \right\} = \left\{ map(\lambda, \Omega_\lhd) \mid \lhd \in ext(T) \right\}$.

Let $\Delta = \left\{ map(\ell, \Omega_\lhd) \mid \lhd \in ext(T_0) \right\}$. We will next show that $\Delta \in \mathbb{S}_\theta^*/\equiv_\theta$ and $\mathsf{ct2lct}(\Delta) = [T_0]$. Fix an arbitrary $u \in \Delta$, from Lemmas 1 and 2, $S_{[u]} = (\Sigma, \prec_0, \sqsubset_0)$. From Theorem 2, $\Delta = \left\{ map(\ell, \Omega_\lhd) \mid \lhd \in ext(S_{[u]}) \right\} = [u]$. And the rest follows. $\square$

The theorem says that the mappings $\mathsf{ct2lct}$ and $\mathsf{lct2ct}$ are inverses of each other and hence are both *bijective*.

5.2 Representation Theorem for lsos-Comtraces and Combined Dependency Graphs

Using Theorem 3, we are going to show that the *combined dependency graph* notion proposed in [26] is another correct alternative definition for comtraces. First we need to define several representation mappings that are needed for our proofs.

Definition 13 (representation mappings $\mathsf{ct2dep}$, $\mathsf{dep2lct}$ and $\mathsf{lct2dep}$). *Let θ be a comtrace alphabet.*

1. *The mapping $\mathsf{ct2dep} : \mathbb{S}_\theta^*/\equiv_\theta \to \mathsf{CDG}(\theta)$ is defined as*
$$
\mathsf{ct2dep}(\mathbf{t}) \triangleq (\Sigma_\mathbf{t}, \prec_u, \sqsubset_u, \ell),
$$
where u is any step sequence in $\mathbf{t}$ and $\prec_u$ and $\sqsubset_u$ are defined as in Definition 5.
2. *The mapping $\mathsf{dep2lct} : \mathsf{CDG}(\theta) \to \mathsf{LCT}(\theta)$ is defined as $\mathsf{dep2lct}(D) \triangleq D^\Diamond$.*
3. *The mapping $\mathsf{lct2dep} : \mathsf{LCT}(\theta) \to \mathsf{CDG}(\theta)$ is defined as*
$$
\mathsf{lct2dep}(T) \triangleq \mathsf{ct2dep} \circ \mathsf{lct2ct}(T). \qquad \blacksquare
$$

Before proceeding futher, we want to make sure that:

Lemma 4. *1. $\mathsf{dep2lct} : \mathsf{CDG}(\theta) \to \mathsf{LCT}(\theta)$ is a well-defined function.*
2. $\mathsf{ct2dep} : \mathbb{S}_\theta^*/\equiv_\theta \to \mathsf{CDG}(\theta)$ is a well-defined function.

Proof. **1.** Given a cd-graph $D_1 = [X, \longrightarrow_1, \dashrightarrow_1, \lambda] \in \mathsf{CDG}(\theta)$, let $T = [X, \prec, \sqsubset, \lambda] = D_1^\Diamond$. We know that T is uniquely defined, since by Definition 11, $(X, \prec, \sqsubset)$ is a so-structure, and so-structures are fixed points of $\Diamond$-closure (by Proposition 2 (4)). We will next show that T is a lsos-comtrace by verifying the conditions $\mathsf{LC1}$–$\mathsf{LC5}$ of Definition 10. Conditions $\mathsf{LC4}$ and $\mathsf{LC5}$ are exactly $\mathsf{CD1}$ and $\mathsf{CD2}$.

LC1: Suppose for contradiction that there exist two distinct non-serializable sets $[\alpha], [\beta] \subset X$ such that $[\alpha](\sqsubseteq^{\mathsf{cov}} \cap \precsim)[\beta]$ and $\lambda([\alpha]) \times \lambda([\beta]) \subseteq ser$. Clearly, this implies that $\alpha \prec \beta$, and thus by the $\lozenge$-closure definition, β is reachable from α on the directed graph $G = (X, \rightsquigarrow)$, where $\rightsquigarrow = \longrightarrow \cup \dashrightarrow$. Now we consider a shortest path P

$$\alpha = \delta_1 \rightsquigarrow \delta_2 \rightsquigarrow \ldots \rightsquigarrow \delta_{k-1} \rightsquigarrow \delta_k = \beta$$

on G that connects α to β. We will prove by induction on $k \geq 2$ that there exist two consecutive δ_i and δ_{i+1} on P such that $\delta_i \in [\alpha]$ and $\delta_{i+1} \in [\beta]$ and $(\lambda(\delta_i), \lambda(\delta_{i+1})) \notin ser$, which contradicts with $\lambda([\alpha]) \times \lambda([\beta]) \subseteq ser$.
Base case: when $k = 2$, then $\alpha \rightsquigarrow \beta$. Since $[\alpha](\sqsubseteq^{\mathsf{cov}} \cap \precsim)[\beta]$, we have $\alpha \longrightarrow \beta$, which by CD3 implies $(\lambda(\alpha), \lambda(\beta)) \notin ser$.
Inductive case: when $k > 2$, we consider δ_1 and δ_2. If $\delta_1 \in [\alpha]$ and $\delta_2 \in [\beta]$, then by $[\alpha](\sqsubseteq^{\mathsf{cov}} \cap \precsim)[\beta]$, we have $\delta_1 \longrightarrow \delta_2$, which immediately yields $(\lambda(\delta_1), \lambda(\delta_2)) \notin ser$. Otherwise, we have $\delta_2 \notin [\alpha] \cup [\beta]$ or $\{\delta_1, \delta_2\} \subseteq [\alpha]$. For the first case, we get $[\alpha]\hat{\sqsubseteq}[\delta_2]\hat{\sqsubseteq}[\beta]$, which contradicts that $[\alpha]\hat{\sqsubseteq}^{\mathsf{cov}}[\beta]$. For the latter case, we can apply induction hypothesis on the path $\delta_2 \rightsquigarrow \ldots \rightsquigarrow \delta_{k-1} \rightsquigarrow \delta_k$.

LC2 and LC3 can also be shown similarly using "shortest path" argument as above. These proofs are easier since we only need to consider paths with edges in $\dashrightarrow$.

2. By the proof of [18, Lemma 4.7], for any two step sequences t and u in $\mathbb{S}_\theta^*$, we have $u \equiv t$ iff $\mathsf{ct2dep}([u]) = \mathsf{ct2dep}([t])$. Thus the mapping $\mathsf{ct2dep}$ is well-defined. $\qquad\square$

Lemma 5. *The mapping* $\mathsf{dep2lct} : \mathsf{CDG}(\theta) \to \mathsf{LCT}(\theta)$ *is injective.*

Proof. Assume that $D_1, D_2 \in \mathsf{CDG}(\theta)$, such that $\mathsf{dep2lct}(D_1) = \mathsf{dep2lct}(D_2) = T = [X, \prec, \sqsubset, \lambda]$. Since $\lozenge$-closure operator does not change the labeling function, we can assume that $D_i = [X, \longrightarrow_i, \dashrightarrow_i, \lambda]$ and $(X, \longrightarrow_i, \dashrightarrow_i)^\lozenge = (X, \prec, \sqsubset)$. We will next show that $(X, \longrightarrow_1, \dashrightarrow_1) \subseteq (X, \longrightarrow_2, \dashrightarrow_2)$.

$(\longrightarrow_1 \subseteq \longrightarrow_2)$: Let $\alpha, \beta \in X$ such that $\alpha \longrightarrow_1 \beta$. Suppose for a contradiction that $\neg(\alpha \longrightarrow_2 \beta)$. Since $\alpha \longrightarrow_1 \beta$, by CD3, $(\lambda(\alpha), \lambda(\beta)) \notin ser$. Thus, by CD2, $\beta \dashrightarrow_2 \alpha$. But since $(X, \longrightarrow_i, \dashrightarrow_i)^\lozenge = (X, \prec, \sqsubset)$, it follows that $(X, \longrightarrow_i, \dashrightarrow_i) \subseteq (X, \prec, \sqsubset)$ (by Proposition 2). Thus, $\alpha \prec \beta$ and $\beta \sqsubset \alpha$, a contradiction.

$(\dashrightarrow_1 \subseteq \dashrightarrow_2)$: Can be proved similarly.

By reversing the role of D_1 and D_2, we have $(X, \longrightarrow_1, \dashrightarrow_1) \supseteq (X, \longrightarrow_2, \dashrightarrow_2)$. Thus, we conclude $D_1 = D_2$. $\qquad\square$

We are now ready to show the following representation theorem which ensures that $\mathsf{lct2dep}$ and $\mathsf{dep2lct}$ are valid representation mappings for $\mathsf{LCT}(\theta)$ and $\mathsf{CDG}(\theta)$.

Theorem 4 (The 2nd Representation Theorem). *Let θ be a comtrace alphabet.*

1. *For every $D \in \mathsf{CDG}(\theta)$, $\mathsf{lct2dep} \circ \mathsf{dep2lct}(D) = D$.*
2. *For every $T \in \mathsf{LCT}(\theta)$, $\mathsf{dep2lct} \circ \mathsf{lct2dep}(T) = T$.*

Proof. **1.** Let $D \in \mathsf{CDG}(\theta)$ and let $T = \mathsf{dep2lct}(D)$. Suppose for a contradiction that $E = \mathsf{lct2dep} \circ \mathsf{dep2lct}(D)$ and $E \neq D$. From how $\mathsf{ct2lct}$ is defined, $\mathsf{ct2lct} = \mathsf{dep2lct} \circ \mathsf{ct2dep}$. Thus, it follows that $\mathsf{dep2lct}(E) = T = \mathsf{dep2lct}(D)$. But this contradicts the injectivity of $\mathsf{dep2lct}$ from Lemma 5.

2. Let $T \in \mathsf{LCT}(\theta)$ and let $D = \mathsf{lct2dep}(T)$. Suppose for a contradiction that $Q = \mathsf{dep2lct} \circ \mathsf{lct2dep}(T)$ and $Q \neq T$. Since $\mathsf{lct2dep} = \mathsf{ct2dep} \circ \mathsf{lct2ct}$, if we let $\mathbf{t} = \mathsf{lct2ct}(T)$, then $Q = \mathsf{dep2lct} \circ \mathsf{ct2dep}(\mathbf{t}) \neq T$. Thus, we have shown that $\mathbf{t} = \mathsf{lct2ct}(T)$ and $\mathsf{ct2lct}(\mathbf{t}) = \mathsf{dep2lct} \circ \mathsf{ct2dep}(\mathbf{t}) \neq T$, contradicting Theorem 3 (2). $\qquad\square$

This theorem shows that lsos-comtraces and cd-graphs are equivalent representations for comtraces. The main advantage of cd-graph definition is its simplicity while the lsos-comtrace definition is stronger and more convenient to prove properties about labeled so-structures that represent comtraces.

We do not need to prove another representation theorem for cd-graphs and comtraces since their representation mappings are simply the composition of the representation mappings from Theorems 3 and 4.

6   Composition Operators

Recall for a comtrace monoid $(\mathbb{S}^*/\!\equiv_\theta, \circledast, [\varepsilon])$, the comtrace operator $\_\circledast\_$ is defined as $[r] \circledast [t] = [r * t]$. We will construct analogous composition operators for lsos-comtraces and cd-graphs. We will then show that lsos-comtraces (cd-graphs) over a comtrace alphabet θ together with its composition operator form a monoid isomorphic to the comtrace monoid $(\mathbb{S}^*/\!\equiv_\theta, \circledast, [\varepsilon])$.

Given two sets X_1 and X_2, we write $X_1 \uplus X_2$ to denote the *disjoint union* of X_1 and X_2. Such disjoint union can be easily obtained by renaming the elements in X_1 and X_2 so that $X_1 \cap X_2 = \emptyset$. We define the lsos-comtrace composition operator as follows.

Definition 14 (composition of lsos-comtraces). *Let T_1 and T_2 be lsos-comtraces over an alphabet $\theta = (E, sim, ser)$, where $T_i = [X_i, \prec_i, \sqsubset_i, \lambda_i]$. The composition $T_1 \odot T_2$ of T_1 and T_2 is defined as (a lp-isomorphic class of) a labeled so-structure $[X, \prec, \sqsubset, \lambda]$ such that $X = X_1 \uplus X_2$, $\lambda = \lambda_1 \cup \lambda_2$, and $(X, \prec, \sqsubset) = \left(X, \prec_{\langle 1,2 \rangle}, \sqsubset_{\langle 1,2 \rangle}\right)^\Diamond$, where*

$$\prec_{\langle 1,2 \rangle} = \prec_1 \cup \prec_2 \cup \left\{ (\alpha, \beta) \in X_1 \times X_2 \mid (\lambda(\alpha), \lambda(\beta)) \notin ser \right\}$$
$$\sqsubset_{\langle 1,2 \rangle} = \sqsubset_1 \cup \sqsubset_2 \cup \left\{ (\alpha, \beta) \in X_1 \times X_2 \mid (\lambda(\beta), \lambda(\alpha)) \notin ser \right\} \qquad\blacksquare$$

Observe that the operator is well-defined since we can easily check that:

Proposition 9. *For every $T_1, T_2 \in \mathsf{LCT}(\theta)$, $T_1 \odot T_2 \in \mathsf{LCT}(\theta)$.* $\qquad\square$

We will next show that this composition operator $\_\odot\_$ properly corresponds to the operator $\_\circledast\_$ of the comtrace monoid over θ.

Proposition 10. *Let θ be a comtrace alphabet. Then*

1. *For every $R, T \in \mathsf{LCT}(\theta)$, $\mathsf{lct2ct}(R \odot T) = \mathsf{lct2ct}(R) \circledast \mathsf{lct2ct}(T)$.*
2. *For every $\mathbf{r}, \mathbf{t} \in \mathbb{S}^*_\theta/\!\equiv_\theta$, $\mathsf{ct2lct}(\mathbf{r} \circledast \mathbf{t}) = \mathsf{ct2lct}(\mathbf{r}) \odot \mathsf{ct2lct}(\mathbf{t})$.*

Proof. **1.** Assume $R = [X_1, \prec_1, \sqsubset_1, \lambda_1]$, $T = [X_2, \prec_2, \sqsubset_2, \lambda_2]$ and $Q = [X_1 \uplus X_1, \prec, \sqsubset, \lambda]$. We can pick $\lhd_1 \in ext(R)$ and $\lhd_2 \in ext(T)$. Then observe that a stratified order $\lhd$ satisfying $\Omega_\lhd = \Omega_{\lhd_1} * \Omega_{\lhd_2}$ is an extension of Q. Thus, by Theorem 3, we have $\mathsf{lct2ct}(R) \circledast \mathsf{lct2ct}(T) = [\mathsf{map}(\lambda_1, \lhd_1)] \circledast [\mathsf{map}(\lambda_2, \lhd_2)] = [\mathsf{map}(\lambda, \lhd)] = \mathsf{lct2ct}(Q)$ as desired.

2. Without loss of generality, we can assume that $\mathbf{r} = [r]$, $\mathbf{t} = [t]$ and $\mathbf{q} = [q] = \mathbf{r} \circledast \mathbf{t}$, where $q = r * t$. By appropriate reindexing, we can also assume that $\Sigma_{\mathbf{q}} = \Sigma_{\mathbf{r}} \uplus \Sigma_{\mathbf{t}}$. Under these assumptions, let $\mathsf{ct2lct}(\mathbf{r}) = T_1 = [\Sigma_{\mathbf{r}}, \prec_{\mathbf{r}}, \sqsubset_{\mathbf{r}}, l_1]$, $\mathsf{ct2lct}(\mathbf{t}) = T_2 = [\Sigma_{\mathbf{t}}, \prec_{\mathbf{t}}, \sqsubset_{\mathbf{t}}, l_2]$ and $\mathsf{ct2lct}(\mathbf{q}) = T = [\Sigma_{\mathbf{q}}, \prec_{\mathbf{q}}, \sqsubset_{\mathbf{q}}, l]$, where $l = l_1 \cup l_2$ is simply the standard labeling functions. It will now suffice to show that $T_1 \odot T_2 = T$.

($\subseteq$): Let $T_1 \odot T_2 = (\Sigma_{\mathbf{r}} \uplus \Sigma_{\mathbf{t}}, \prec_{\langle \mathbf{r}, \mathbf{t} \rangle}, \sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle}, l)^{\Diamond}$. By Definitions 5 and 7, we have

$$\prec_{\langle \mathbf{r}, \mathbf{t} \rangle} = \prec_{\mathbf{r}} \cup \prec_{\mathbf{t}} \cup \{(\alpha, \beta) \in \Sigma_{\mathbf{r}} \times \Sigma_{\mathbf{t}} \mid (\lambda(\alpha), \lambda(\beta)) \notin ser\} \subseteq \prec_{\mathbf{q}}$$
$$\sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle} = \sqsubset_{\mathbf{r}} \cup \sqsubset_{\mathbf{t}} \cup \{(\alpha, \beta) \in \Sigma_{\mathbf{r}} \times \Sigma_{\mathbf{t}} \mid (\lambda(\beta), \lambda(\alpha)) \notin ser\} \subseteq \sqsubset_{\mathbf{q}}$$

Thus, by Proposition 2 (5), we have $(\Sigma_{\mathbf{r}} \uplus \Sigma_{\mathbf{t}}, \prec_{\langle \mathbf{r}, \mathbf{t} \rangle}, \sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle}, l)^{\Diamond} \subseteq (\Sigma_{\mathbf{q}}, \prec_{\mathbf{q}}, \sqsubset_{\mathbf{q}}, l)$ as desired. Furthermore, by Proposition 2 (5), $(\Sigma_{\mathbf{r}} \uplus \Sigma_{\mathbf{t}}, \prec_{\langle \mathbf{r}, \mathbf{t} \rangle}, \sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle})^{\Diamond}$ is a so-structure.

($\supseteq$): By Definitions 5 and 7, we have $\prec_q \subseteq \prec_{\langle \mathbf{r}, \mathbf{t} \rangle}$ and $\sqsubset_q \subseteq \sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle}$. Since we already know $(\Sigma_{\mathbf{r}} \uplus \Sigma_{\mathbf{t}}, \prec_{\langle \mathbf{r}, \mathbf{t} \rangle}, \sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle})^{\Diamond}$ is a so-structure, it follows from Proposition 2 (5) that

$$(\Sigma_{\mathbf{q}}, \prec_{\mathbf{q}}, \sqsubset_{\mathbf{q}}, l) = (\Sigma_{\mathbf{q}}, \prec_q, \sqsubset_q, l)^{\Diamond} \subseteq (\Sigma_{\mathbf{r}} \uplus \Sigma_{\mathbf{t}}, \prec_{\langle \mathbf{r}, \mathbf{t} \rangle}, \sqsubset_{\langle \mathbf{r}, \mathbf{t} \rangle}, l)^{\Diamond} = T_1 \odot T_2. \qquad \square$$

Let $\mathbb{I}$ denote the lp-isomorphic class $[\emptyset, \emptyset, \emptyset, \emptyset]$. Then we observe that $\mathsf{ct2lct}([\varepsilon]) = \mathbb{I}$ and $\mathsf{lct2ct}(\mathbb{I}) = [\varepsilon]$. By Proposition 10 and Theorem 3, the structure $(\mathsf{LCT}(\theta), \odot, \mathbb{I})$ is isomorphic to the monoid $(\mathbb{S}_{\theta}^* / \equiv_{\theta}, \circledast, [\varepsilon])$ under the isomorphisms $\mathsf{ct2lct} : \mathbb{S}_{\theta}^* / \equiv_{\theta} \to \mathsf{LCT}(\theta)$ and $\mathsf{lct2ct} : \mathsf{LCT}(\theta) \to \mathbb{S}_{\theta}^* / \equiv_{\theta}$. Thus, the triple $(\mathsf{LCT}(\theta), \odot, \mathbb{I})$ is also a monoid. We can summarize these facts in the following theorem:

Theorem 5. *The mappings* $\mathsf{ct2lct}$ *and* $\mathsf{lct2ct}$ *are monoid isomorphisms between two monoids* $(\mathbb{S}_{\theta}^* / \equiv_{\theta}, \circledast, [\varepsilon])$ *and* $(\mathsf{LCT}(\theta), \odot, \mathbb{I})$. $\qquad \square$

Similarly, we can also define a composition operator for cd-graphs.

Definition 15 (composition of cd-graphs). *Let D_1 and D_2 be cd-graphs over an alphabet $\theta = (E, sim, ser)$, where $D_i = [X_i, \longrightarrow_i, \dashrightarrow_i, \lambda_i]$. The composition $D_1 \circledcirc D_2$ of D_1 and D_2 is defined as (a lp-isomorphic class of) a labeled so-structure $[X, \longrightarrow, \dashrightarrow, \lambda]$ such that $X = X_1 \uplus X_2$, $\lambda = \lambda_1 \cup \lambda_2$, and*

$$\longrightarrow = \longrightarrow_1 \cup \longrightarrow_2 \cup \{(\alpha, \beta) \in X_1 \times X_2 \mid (\lambda(\alpha), \lambda(\beta)) \notin ser\}$$
$$\dashrightarrow = \dashrightarrow_1 \cup \dashrightarrow_2 \cup \{(\alpha, \beta) \in X_1 \times X_2 \mid (\lambda(\beta), \lambda(\alpha)) \notin ser\} \qquad \blacksquare$$

From this definition, it is straightforward to show the following propositions, which we will state without proofs.

Proposition 11. *For every $D_1, D_2 \in \mathsf{CDG}(\theta)$, $D_1 \circledcirc D_2 \in \mathsf{CDG}(\theta)$.* $\qquad \square$

Proposition 12. *Let θ be a comtrace alphabet. Then*

1. *For every $R, T \in \mathsf{LCT}(\theta)$, $\mathsf{lct2dep}(R \odot T) = \mathsf{lct2dep}(R) \circledcirc \mathsf{lct2dep}(T)$.*
2. *For every $D, E \in \mathsf{CDG}(\theta)$, $\mathsf{dep2lct}(D \circledcirc E) = \mathsf{dep2lct}(D) \odot \mathsf{dep2lct}(E)$.* $\qquad \square$

Putting the two preceding propositions and Theorem 4 together, we conclude:

Theorem 6. *The mappings* $\mathsf{lct2dep}$ *and* $\mathsf{dep2lct}$ *are monoid isomorphisms between two monoids* $(\mathsf{LCT}(\theta), \odot, \mathbb{I})$ *and* $(\mathsf{CDG}(\theta), \circledcirc, \mathbb{I})$. $\qquad \square$

7 Conclusion

The simple yet useful construction we used extensively in this paper is to build a quotient so-structure modulo the ⊏-cycle equivalence relation. Intuitively, each ⊏-cycle equivalence class consists of all the events that must be executed simultaneously with one another and hence can be seen as a single "composite event". The resulting quotient so-structure is technically easier to handle since both relations of the quotient so-structure are acyclic. From this construction, we were able to give a labeled so-structure definition for comtraces similar to the labeled poset definition for traces. This quotient construction also explicitly reveals the following connection: a step on a step sequence *s* is not serializable with respect to the relation *ser* of a comtrace alphabet if and only if it corresponds to a ⊏-cycle equivalence class of the lsos-comtrace representing the comtrace [*s*] (cf. Proposition 4).

We have also formally shown that the quotient monoid of comtraces, the monoid of lsos-comtraces and the monoid of cd-graphs *over the same comtrace alphabet* are indeed isomorphic by establishing monoid isomorphisms between them. These three models are formal linguistic, order-theoretic, and graph-theoretic respectively, which allows us to apply a variety of tools and techniques.

An immediate future task is to develop a framework similar to the one in this paper for *generalized comtraces*, proposed and developed in [21,27,22]. Generalized comtraces extend comtraces with the ability to model events that can be executed *earlier than or later than but never simultaneously*. Another direction is to define and analyze infinite comtraces (and generalized comtraces) in a spirit similar to the works on infinite traces, e.g., [13,4]. It is also promising to use infinite lsos-comtraces and cd-graphs to develop logics for comtraces similarly to what have been done for traces (cf. [31,7]).

References

1. E. Clarke, O. Grumberg and D. Peled, *Model Checking*, MIT Press, Cambridge, 1999.
2. T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to Algorithms*, Second Edition, MIT Press, 2001.
3. B. A. Davey and H. A. Priestley, *Introduction to Lattices and Order*, Cambridge University Press 2002.
4. V. Diekert, On the Concatenation of Infinite Traces, Proc. of STACS, *LNCS* 480 (1991), 105–117.
5. V. Diekert and G. Rozenberg (eds.), *The Book of Traces*. World Scientific 1995.
6. V. Diekert and Y Métivier, *Partial Commutation and Traces*, *Handbook of Formal Languages, Vol. 3: Beyond Words*, pp 457 - 533, Springer 1997.

7. V. Diekert, M. Horsch, M. Kufleitner, On First-Order Fragments for Mazurkiewicz Traces, *Fundam. Inform.* 80(1-3): 1-29, 2007.

8. J. Esparza and K. Heljanko, *Unfoldings – A Partial-Order Approach to Model Checking*, Springer 2008.

9. A. Farzan and P. Madhusudan, Causal Dataflow Analysis for Concurrent Programs, Proc. of CAV, *LNCS* 4144 (2006): 315–328.

10. A. Farzan and P. Madhusudan, Causal Atomicity, Proc. of TACAS 2007, *LNCS* 4424 (2007), 102–116.

11. P. C. Fishburn, *Interval Orders and Interval Graphs*, J. Wiley 1985, New York.

12. H. Gaifman and V. Pratt, Partial Order Models of Concurrency and the Computation of Function, *Proc. of LICS'87*, pp. 72–85.

13. P. Gastin, Infinite Traces, Proc. of Semantics of Systems of Concurrent Processes, *LNCS* 469 (1990), 277–308.

14. T. Gazagnaire, B. Genest, L. Hélouët, P. S. Thiagarajan, S. Yang, Causal Message Sequence Charts, *Theor. Comput. Sci.* 410(41): 4094–4110, 2009.

15. R. Janicki, Relational Structures Model of Concurrency. *Acta Informatica*, 45(4): 279–320, 2008.

16. R. Janicki and M. Koutny, Invariants and Paradigms of Concurrency Theory, Proc. of *PARLE '91*, *LNCS* 506, Springer 1991, pp. 59–74.

17. R. Janicki and M. Koutny, Structure of Concurrency, *Theoretical Computer Science*, 112(1): 5–52, 1993.

18. R. Janicki and M. Koutny, Semantics of Inhibitor Nets, *Information and Computation*, 123(1): 1–16, 1995.

19. R. Janicki and M. Koutny, Fundamentals of Modelling Concurrency Using Discrete Relational Structures, *Acta Informatica*, 34: 367–388, 1997.

20. R. Janicki and M. Koutny, On Causality Semantics of Nets with Priorities, *Fundamenta Informaticae* 34: 222–255, 1999.

21. R. Janicki and D. T. M. Lê, Modelling Concurrency with Quotient Monoids, Proc of PETRI NETS 2008, *LNCS* 5062, Springer 2008, pp. 251–269.

22. R. Janicki and D. T. M. Lê, Modelling Concurrency with Comtraces and Generalized Comtraces, submitted in 2009. Available at: http://arxiv.org/abs/0907.1722

23. G. Juhás, R. Lorenz, S. Mauser, Causal Semantics of Algebraic Petri Nets distinguishing Concurrency and Synchronicity, *Fundamenta Informatica* 86(3): 255-298, 2008.

24. G. Juhás, R. Lorenz, S. Mauser, Synchronous + Concurrent + Sequential = Earlier Than + Not Later Than, Proc. of ACSD'06, Turku, Finland 2006, pp. 261-272, IEEE Press.

25. H. C. M. Kleijn and M. Koutny, Process Semantics of General Inhibitor Nets, *Information and Computation*, 190:18–69, 2004.

26. J. Kleijn and M. Koutny, Formal Languages and Concurrent Behaviour, *Studies in Computational Intelligence*, 113:125-182, 2008.

27. D. T. M. Lê, Studies in Comtrace Monoids, Master Thesis, Dept. of Computing and Software, McMaster University, Canada, August 2008.

28. A. Mazurkiewicz, Concurrent Program Schemes and Their Interpretation, TR DAIMI PB-78, Comp. Science Depart., Aarhus University, 1977.

29. V. Pratt, Modeling concurrency with partial orders, *International Journal of Parallel Programming*, 15(1):33–71, 1986.

30. E. Szpilrajn, Sur l'extension de l'ordre partiel, *Fund. Mathematicae* 16, 386–389, 1930.

31. P. S. Thiagarajan and I. Walukiewicz, An expressively complete linear time temporal logic for Mazurkiewicz traces, *Inf. Comput.* 179(2): 230–249, 2002.

A Proof of Proposition 4

Proof. **1.** ($\Rightarrow$): Since $[\alpha] = [\beta]$, we know that $\alpha = \beta$ or ($\alpha \sqsubset \beta \wedge \beta \sqsubset \alpha$). The former case is trivial. For the latter case, by Theorem 1, we have $\forall \lhd \in ext(S).\ \alpha \lhd^\frown \beta$ and $\forall \lhd \in ext(S).\ \beta \lhd^\frown \alpha$. But this implies that $\forall \lhd \in ext(S).\ \alpha \frown_\lhd \beta$.

($\Leftarrow$): The case when $\alpha = \beta$ is trivial. Assume that $\alpha \neq \beta$ and $\forall \lhd \in ext(S).\ \alpha \frown_\lhd \beta$. Thus, by Theorem 1, $\alpha \sqsubset \beta$ and $\beta \sqsubset \alpha$. But this means α and β belong to the same equivalence class.

2. Suppose for a contradiction that all $\lhd \in ext(S)$ cannot be written in the form of $\Omega_\lhd = u[\alpha]v$. This implies that there exists some $\gamma \in X \setminus [\alpha]$ such that for all $\lhd \in ext(S)$, $\alpha \frown_\lhd \gamma$. But by Theorem 1, this yields $\alpha \sqsubset \gamma$ and $\gamma \sqsubset \alpha$, contradicting with $\gamma \notin [\alpha]$.

3. Assume $[\alpha] \hat{\sqsubset}^{cov} [\beta]$. Suppose for a contradiction that there does not exist $\lhd \in ext(S)$ such that $\Omega_\lhd = u[\alpha][\beta]v$ for some step sequences u and v. Then, by Theorem 1, there must exist some $\gamma \in X \setminus ([\alpha] \cup [\beta])$, such that $\alpha \sqsubset \gamma \sqsubset \beta$. Since $\gamma \notin [\alpha] \cup [\beta]$, this yields $[\alpha] \sqsubset [\gamma] \sqsubset [\beta]$, which contradicts that $[\alpha] \hat{\sqsubset}^{cov} [\beta]$. $\qquad\square$

B Proof of Proposition 5

Proof. **1.** Assume $\alpha, \beta \in A_i$ and $\alpha \neq \beta$. Thus, $\alpha \frown_\lhd \beta$. Thus, by Corollary 1 (2), we have $\alpha \frown_\prec \beta$. Hence, by **LC5** of Definition 10, $(\lambda(\alpha), \lambda(\beta)) \in sim$. Since *sim* is irreflexive, this also shows that any two distinct α and β in A_i have different labels. Thus, $|A_i| = |\lambda(A_i)|$ for all i.

2. From the proof of **1.**, we know that $\alpha, \beta \in A_i$ and $\alpha \neq \beta$ implies $(\lambda(\alpha), \lambda(\beta)) \in sim$. Thus, $\lambda(A_i) \in \mathbb{S}_\theta$ for all i. $\qquad\square$

C Proof of Proposition 6

Proof. Observe that from Proposition 5, we have $\Sigma_v = \Sigma_w$. It remains to show that $\xi_v(\alpha) = \xi_w(\alpha)$ for all $\alpha \in \Sigma_v$.

Suppose for a contradiction that $\xi_v(\alpha) \neq \xi_w(\alpha)$ for some $\alpha \in \Sigma_v$. From the definition above, there are two distinct elements $x, y \in X$, such that $\xi_v(\alpha) = x$ and $\xi_w(\alpha) = y$ and $\lambda(x) = \lambda(y) = \ell(\alpha)$. Since *ser* is irreflexive, $(\lambda(x), \lambda(y)) \notin ser \cup ser^{-1}$. Thus, by **LC4** of Definition 10, $x \prec y$ or $y \prec x$. Without loss of generality, we assume $x \prec y$ and that $\alpha = a^{(i)}$ for some event $a \in E$.

Again by **LC4** of Definition 10, we know that elements having the same label are totally ordered by $\prec$. Thus, if k is the number of elements in X labeled by a, then we have $\xi_w(a^{(1)}) \prec \xi_w(a^{(2)}) \prec \ldots \prec \xi_w(a^{(k)})$ and $\xi_v(a^{(1)}) \prec \xi_v(a^{(2)}) \prec \ldots \prec \xi_v(a^{(k)})$. But then $\xi_v(a^{(i)}) = x$ implies that $|\{z \in X \mid z \prec y \wedge \lambda(z) = a\}| \geq i$, while $\xi_w(a^{(i)}) = y$ implies that $|\{z \in X \mid z \prec y \wedge \lambda(z) = a\}| < i$, which is absurd. $\qquad\square$

D Proof of Lemma 3

Proof. Let $T = (\Sigma_{[u]}, \prec_{[u]}, \sqsubset_{[u]}, l)$. From Theorem 2, T is a labeled so-structure. It only remains to show that T satisfies conditions **LC1**–**LC5** of Definition 10.

LC1: Assume $[\alpha](\hat{\sqsubseteq}^{\mathsf{cov}} \cap \hat{\succsim})[\beta]$ and suppose for a contradiction that $\lambda([\alpha]) \times \lambda([\beta]) \subseteq ser$. Then from Proposition 4 (3), there exists $\lhd \in ext(T)$ such that $\Omega_{\lhd} = v[\alpha][\beta]w$. From Theorem 2, since we have $\lhd \in \{\lhd_s \mid s \in [u]\} = ext(S_{[u]})$, it follows that $\mathsf{map}(l, v[\alpha][\beta]w) \in [u]$. But $\lambda([\alpha]) \times \lambda([\beta]) \subseteq ser$ implies $\mathsf{map}(\ell, u[\alpha] \cup [\beta]v) \in [u]$. Hence, $u[\alpha] \cup [\beta]v$ is also a stratified extension of T, which contradicts that $[\alpha]\hat{\succsim}[\beta]$. Using a similar argument, we can show LC2 using Proposition 4 (1,4) and LC3 using Proposition 4 (1,2).

LC4: Follows from Definitions 5 and 7 and the $\lozenge$-closure definition.

LC5: Since $\alpha \frown_{\prec} \beta$, it follows from Corollary 1 that there exists $\lhd \in ext(T)$ where $\alpha \frown_{\lhd} \beta$. Since $\{\lhd_s \mid s \in [u]\} = ext(S_{[u]})$, there exists a sequence $s \in [u]$ such that $s = \mathsf{map}(f, \Omega_{\lhd})$. This implies α and β belong to the same step in $\bar{s}$. Thus, we have $(\lambda(\alpha), \lambda(\beta)) \in sim$. $\qquad\square$